Problem 1

Search Algorithm	Optimal	Expansions	Path Length	Execution Time (s)
Breadth First Search	Yes	43	6	0.00594411
Depth First Graph Search	No	21	20	0.00303032
Uniform cost Search	Yes	60	6	0.00894590
Greedy Best First Graph Search 1	Yes	7	6	0.00164158
Greedy Best First Graph Search 2	Yes	6	6	0.41193465
Greedy Best First Graph Search 3	Yes	6	6	0.29929283
Greedy Best First Graph Search 4	Yes	6	6	0.50776247
A Star Search 1	Yes	50	6	0.00890677
A Star Search 2	Yes	28	6	1.02987356
A Star Search 3	Yes	43	6	1.05396838
A Star Search 4	Yes	33	6	1.21358165

Problem 2:

Search Algorithm	**Optimal**	**Expansions**	**Path Length**	**Execution Time (s)**
Breadth First Search	Yes	3343	9	1.81303950
Depth First Search	No	624	619	2.78556195
Uniform Cost Search	Yes	5154	9	3.08115778
Greedy Best First Search 1	Yes	17	9	0.01767745
Greedy Best First Search 2	Yes	9	9	9.00224264
Greedy Best First Search 3	Yes	27	9	18.50740043
Greedy Best First Search 4	Yes	9	9	12.47610014
A Star Search 1	Yes	2467	9	2.095102238
A Star Search 2	Yes	357	9	227.82771191
A Star Search 3	Yes	2887	9	1228.34163389
A Star Search 4	Yes	1037	9	1042.03801595

Problem 3

Search Algorithm	Optimal	Expansions	Path Length	Execution Time (s)
Breadth First Search	Yes	14663	12	8.53046281
Depth First Search	No	408	392	0.98303380
Uniform Cost Search	Yes	18512	12	11.76837428
Greedy Best First Search 1	No	25	15	0.03065162
Greedy Best First Search 2	No	14	14	18.51682551
A Star Search 1	Yes	7388	12	6.90584898
A Star Search 2	Yes	369	12	339.75939135

Problem 4:

Search Algorithm	Optimal	Expansions	Path Length	Execution Time (s)
Breadth First Search	Yes	99736	14	76.85300303
Depth First Search	-	-	-	-
Uniform Cost Search	Yes	113339	14	93..64420320
Greedy Best First Search 1	No	29	18	0.04785316
Greedy Best First Search 2	No	17	17	33.73382394
A Star Search 1	Yes	34330	14	44.22115365

A Star Search 2	No	1208	15	1877.04277142

Number of Nodes Expanded vs. Number of Actions

Algorithm	Action-problem 1	Nodes-problem 1	Action-problem 2	Action-problem 2
BFS	20	43	72	3343
DFS	20	21	72	624
UCS	20	60	72	5154
GBFS 1	20	7	72	17
GBFS 2	20	6	72	9
GBFS 3	20	6	72	27
GBFS 4	20	6	72	9
A* Search 1	20	50	72	2467
A* Search 2	20	28	72	357
A* Search 3	20	43	72	2887
A* Search 4	20	33	72	1037

Algorithm	Action-problem 3	Nodes-problem 3	Action-problem 4	Nodes-problem 4
BFS	88	14663	104	99736
DFS	88	408	104	-
UCS	88	18512	104	113339
GBFS 1	88	25	104	29
GBFS 2	88	14	104	17
A* Search 1	88	7388	104	34330

A* Search 2	88	369	104	1208

Search Time vs. Number of Actions

Algorithm	Search Time 1	Actions Problem 1	Search Time 2	Actions Problem 2
BFS	0.00594411	20	1.81303950	72
DFS	0.00303032	20	2.78556195	72
UCS	0.00894590	20	3.08115778	72
GBFS 1	0.00164158	20	0.01767745	72
GBFS 2	0.41193465	20	9.00224264	72
GBFS 3	0.29929283	20	18.50740043	72
GBFS 4	0.50776247	20	12.47610014	72
A* Search 1	0.00890677	20	2.095102238	72
A* Search 2	1.02987356	20	227.82771191	72
A* Search 3	1.05396838	20	1228.34163389	72
A* Search 4	1.21358165	20	1042.03801595	72

Algorithm	Search Time 3	Actions Problem 3	Search Time 4	Actions Problem 4
BFS	8.53046281	88	76.85300303	104
DFS	0.98303380	88	-	104
UCS	11.76837428	88	93..64420320	104
GBFS 1	0.03065162	88	0.04785316	104
GBFS 2	18.51682551	88	33.73382394	104
A* Search 1	6.90584898	88	44.22115365	104

A* Search 2	339.75939135	88	1877.04277142	104

Questions

1. Which algorithms are suitable for planning in restricted domains?

Air cargo problem number 1’s conditions present an example of a restricted domain. Given that we want to achieve the lowest execution time, greedy best first search and depth first search are both useful algorithms for doing this.

2. Which algorithms are suitable for planning in large domains?

Air cargo problem number 4’s conditions would resemble a larger domain. In this case, greedy best first search does the best job of achieving the lowest execution time!

3. Which algorithms are better in case of optimal plans?

As shown in the figure for problem 4, breadth first search would be the best algorithm to plan the optimal path. This is because it is able to do so in the least amount of time compared to uniform cost search and A* search. Depth first search is unable to find the optimal path, so it is not considered.